A Novel Area Efficient Encoder/Decoder Design for Crosstalk Noise Elimination in SOC/ASIC

Vidya Patel.N [1], D.R.Shilpa[2], Dr.B.V.Uma[3]

Abstract - Most of the encoding methods proposed in recent years have dealt with only RC modeled VLSI interconnects. For deep submicron technologies (DSM), on-chip inductive effects have increased due to faster clock speeds, smaller signal rise times and longer length of on-chip interconnects. All these issues raise the concern for crosstalk, propagation delay and power dissipation of overall. Therefore, this paper introduces an efficient Bus Encoder using Bus Inverting (BI) method. The proposed design dramatically reduces both crosstalk and power dissipation in RLC modeled interconnects which makes it suitable for current high-speed low-power VLSI interconnects. The proposed model demonstrates an overall reduction in the number of transistors used by 30%.

Index terms - Bus-invert, Crosstalk, Coupling, Decoder, Encoder, Power dissipation, Signal integrity, switching activity**.**

1. INTRODUCTION

The performance of a high-speed chip in a deep sub-micron technology is largely dependent on interconnects which connect different macro cells within a VLSI/ULSI chip [1]. As device geometries shrink, chip sizes increase, and clock speeds get faster, interconnect delay is becoming increasingly significant. Signal integrity is the ability of an electrical signal to carry information reliably and resist the effects of high-frequency electromagnetic interference from nearby signals. Crosstalk is the undesirable electrical interaction between two or more physically adjacent nets due to capacitive cross-coupling. As integrated circuit technologies advance toward smaller geometries, crosstalk effects become increasingly important compared to cell delays and net delays.

In particular, the propagation delay through long cross-chip buses is already proving to be a limiting factor in the speed of some designs, and this trend will only get worse. It has been shown that the delay through a long bus is strongly a function of the coupling capacitance between the wires.

Especially detrimental to the delay is the Miller-like effect when adjacent wires simultaneously transition in opposite directions. When the cross-coupling capacitance is comparable to or exceeds the loading capacitance on the wires, the delay of such a transition may be twice or more that of a wire transitioning next to a steady signal. We call this delay penalty the “crosstalk delay”.

There are different methods for the reduction of crosstalk such as repeater insertion, shielding line (Vdd /GND) insertion between two adjacent wires [2], optimal spacing between signal lines and lastly the most effective Bus encoding method [3]-[8]. This paper uses bus invert method for the reduction of power dissipation, crosstalk, propagation delay and chip size of encoder and decoder of RLC modeled interconnects.A bus encoding method converts or encode data bit stream in such a manner so that the transitions of bit stream are minimized. This method reduces power and crosstalk delay by decreasing switching and coupling activities. The proposed method reduces undesirable types of crosstalk i.e., Type-0, Type-1, Type-4, Type-3 and some Type-2 coupling which are worst case scenarios observed in RLC type of interconnects. Furthermore, the proposed design reduces the power dissipation by reducing switching activity as compared with Fan et al. [9].

The rest of the paper is organized as follows. The section 1 introduces to the critical research scenario in VLSI interconnects. Section 2 describes crosstalk

. Vidya Patel.N is currently pursuing M.Tech in VLSI design and Embedded systems in RVCE, Bangalore, India. Email: vidyaptl91@gmail.com

. D.R.Shilpa is currently working as Assistant Professor in Department of Electronics and Communication, RVCE, Bangalore, India. Email: shilpadr@rvce.edu.in

. Dr. B.V.Uma is currently working as Professor and Dean in Department of Electronics and communication, RVCE, Bangalore, India. Email: umabv@rvce.edu.in

and power dissipation expression and their dependency on different parameters. Section 3 describes the working of proposed method. Section 4 discusses the results obtained for encoder. Finally, section 5 draws important conclusion.

2. POWER AND CROSSTALK IN RLC MODELLED INTERCONNECT

The parasitic capacitance model of an interconnect consists of three parts, the ground capacitance (C_G), the fringe or side-wall capacitance to substrate (C_F) and coupling capacitance (C_C).

Coupling capacitance becomes dominant when adjacent wire tend to switch from 1 to 0 or 0 to 1, resulting in delay penalty which is called crosstalk delay. There are two important effects due to this crosstalk i.e., noise on non-switching wires and increased delay on switching wires.

Assume two lines namely A and B and their associated capacitances. According to the behaviour of neighbouring wires, effective coupling capacitance (C_{eff}) can be evaluated. Table 1 shows the dependence of effective capacitance of line A ($C_{eff(A)}$) on line B (assuming line A is switching).

Table 1: Dependence of Effective Capacitance of Line 'A' on Line 'B'

Line 'B'	ΔV	$C_{eff(A)}$	Miller Coupling Factor(MCF)
Switching with 'A'	0	C_G	0
Constant	V_{dd}	C_C+ C_G	1
Switching oppositely with 'A'	$2V_{dd}$	$2C_C$+ C_G	2

Firstly, it is observed that when both adjacent lines are switching in the same direction the Miller Coupling Factor (MCF) is '0' which indicates that there is no coupling capacitance. Secondly, if one line is switching and the other is quiet then MCF is '1' whose coupling capacitance is greater than the case of MCF=0. Finally, when two adjacent lines are switching in opposite direction then the coupling capacitance is highest (MCF is '2') due to which crosstalk effect becomes dominant. In a data bus there will be adjacent lines to the left and right side of the line. Therefore, various coupling capacitances associated with the 3-bit configuration must be considered. Line B is line of interest whereas A and C are adjacent lines to it. Coupling factor associated with line B depends on the switching configurations of lines A and C.

All possible switching configurations can be classified to Type-0, Type-1, Type-2, Type-3 and Type-4 depending on the value of MCF as shown in the Table 2.

Table 2: Classification of Crosstalk

Type-0	Type-1	Type-2	Type-3	Type-4
−−−	−−↑	−↑−	−↑↓	↑↓↑
↑↑↑	−↑↑	↑−↑	−↓↑	↓↑↓
↓↓↓	↑−−	↑−↓	↑↓−	
	↑↑−	↑↑↓	↓↑−	
	−−↓	↑↓↓		
	−↓↓	−↓−		
	↓−−	↓−↓		
	↓↓−	↓−↑		
		↓↓↑		
		↓↑↑		

↑: Switching from 0 to 1, ↓: Switching from 1 to 0, −: no transition

Type-0 coupling occurs when present data and the previous data bits have transition in all bit positions i.e. (from 000 to 111) or (from 111 to 000).Three conditions are causes Type-0 coupling. In this, coupling capacitance is zero but the mutual inductance is very high. Type-1 coupling occurs when there is one or two transition in the same direction i.e. present data and the previous data have same transition in one or two bit positions (i.e. transitions from 000 to 011) or (from 110 to 111). Eight conditions causes Type-1 coupling. In Type-1 simultaneous transition of two bits causes coupling are lesser as comparison with Type-0. A Type-2 coupling occurs if the centre wire is having opposite transition with one of its adjacent wires (from 011 to 100) or when the other lines undergo the same state transition with the centre wire as quite (i.e. data change from 100 to 001). Ten conditions causes Type-2 coupling. A Type-3 coupling occurs when the centre wire undergoes opposite transition with one of the adjacent wire while the other wires are quite i.e.

when the data changes from 010 to 001. In this case the mutual inductance is very less. In Type-4 coupling, all three wires are having transition in opposite direction with respect to each other i.e. from 010 to 101. Here mutual inductance coupling is zero but it is the worst case of RC model.

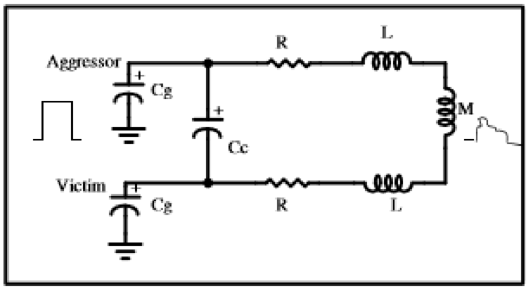

Figure 1: RLC equivalent of an interconnect

Power dissipation is expressed as [9]:

$$P=\alpha*V_{dd}^{2}*f*C_{L} \qquad (1)$$

Where C_L is load capacitance, V_{dd} is supply voltage, f is the clock frequency and α is the average switching activity which lies between 0 and 1.

For reducing power dissipation in VLSI circuits one or more factors such as V_{dd}, f, C_L and α must be minimized. Here, V_{dd} and f are assumed to be already optimized for low power. Therefore, dynamic power dissipation is proportional to the number of signal transition. Symbols and terminologies used throughout this paper are as follows:

d(t) : Bus value at the input of encoder.

D(t): Encoder output which is transmitted

D(t-1): Encoder output which was latched up.

inv(t) : Invert line at the input of encoder which is preset to '0'.

INV(t) : Invert line for the encoded data sent at time t.

INV(t-1) : Invert line for encoded data sent at time t-1.

3. IMPLEMENTATION OF PROPOSED DESIGN

For Type-0 and Type-1:

The proposed encoder is a modified and improved version of the design proposed by [10] and [11] to reduce crosstalk, delay, and the power dissipation of RLC modelled interconnect.

In this proposed method the data bus is divided into different clusters. Each cluster has 4-bit width with one extra control bit. Basically, bus invert method [2, 3] utilizes an extra control bit i.e., INV(t) to differentiate the transmission of original data and inverted data. In this method, if the number of transitions are more than half of the size of bus width, then original data is inverted and control line (INV(t)) is set to 'high' whereas in other case original data is transmitted with INV(t) at logic 'low'.

The block diagram of proposed encoder is shown in figure-2. The 5-bit bus encoder architecture is composed of inverter, CNT0, CNT1_1, CNT1_2, 2-bit comparator, XOR stack and latch. CNT0 and CNT1 (CNT1_1 and CNT1_2) are crosstalk modules used to count Type-0 and Type-1 couplings respectively. Two type-1 counters viz., CNT1_1 and CNT1_2 are used which counts the number of type-1 couplings with original data and inverted data respectively. Brief description of the block diagram is as follows.

The data to be transmitted (d(t),inv(t)) through encoder is simultaneously inverted to obtain $(\overline{d(t)},\overline{inv(t)})$ as shown in Fig.2. Here the value of inv(t) in the data to be transmitted is assumed to be at logic 'low' initially. Now, the original data (d(t),inv(t)) and the previously latched data (D(t-1), INV(t-1)) are fed as inputs to CNT0 and CNT1_1 counters. The outputs of CNT0 (1 bit) and CNT1_1 (2 bits) are N0 and K1K0 respectively. The inverted data $(\overline{d(t)},\overline{inv(t)})$ and the data stored (D(t-1), INV(t-1)) are fed as inputs to CNT1_2 whose output (i.e., of 2 bits) is L1L0. The counts of two type-1 counters are compared in 2-bit comparator. The inputs of 2-bit comparator are K1K0 and L1L0 (which are having 2-bits) whereas the output of the comparator is N1 (1-bit). Next, N0 and N1 are fed as inputs to an OR gate whose output is INV(t) . This INV(t) and the original data (d(t),inv(t)) are given as inputs to XOR stack. The output of XOR stack can be inverted data (if INV(t) is '1') or the original data (if INV(t) is '0'). The output of the XOR stack is the encoded data (D(t), INV(t)) which is finally fed to interconnects. This encoded data is stored in latch for one clock cycle (D(t-1),INV(t-1)), after which it is fed back for comparison with (d(t), inv(t)). Finally, at the receiving side, decoder retrieves the original data with the help of INV(t) line.

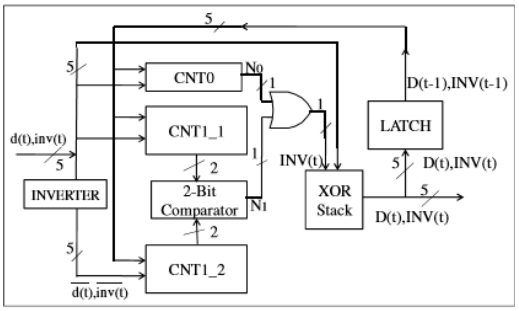

Figure 2: Block diagram of 5-bit bus encoder

CNT0: Type-0 coupling occurs if three adjacent lines have the same transition (i.e., ↑↑↑ or ↓↓↓). CNT0 counts the number of type-0 couplings whose internal circuit diagram is shown in Figure 3. There are two type-0 couplings (one is of 'low to high' transition and the other is of 'high to low' transition) which can be merged to only one coupling. As 'low to high' transition (↑) and 'high to low' transition (↓) are detected separately, it is concluded that there is only one type-0 coupling. First, the design checks the occurrence of transition by using level-1 AND gates. The top five AND gates detect 'high to low' transition whereas bottom five will detect 'low to high' transition. A 'high' logic is present at output if there is a transition (if not logic 'low' is present). The output signals from level-1 AND gates (1 to 5) ('high to low' transition) signals are grouped in to three different combinations as (123, 234, 345). These signals are fed to three different AND gates to detect ↓↓↓ condition as shown in Fig.3. Similarly, the output signals of 'low to high' detector (6, 7, 8, 9, 10) are given to another set of AND gates to detect ↑↑↑ condition. Finally, all these signals are fed to OR gate, whose output (N0) becomes 'high' if any one of the combination satisfies the type-0 coupling condition.

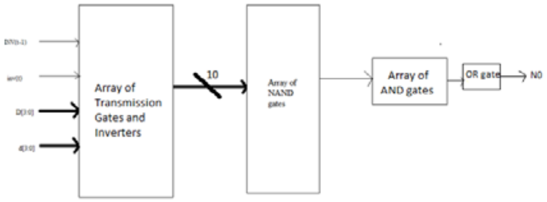

Figure 3: Circuit diagram of CNT0

CNT1_1: CNT1_1 Type-1 coupling occurs when one or two lines are having transition in the same direction while the rest (i.e., remaining two or the third one respectively) are idle. There are eight conditions of type-1 coupling which can be placed in two different groups with each group having four switching conditions that depends on high to low and low to high transitions. CNT1_1 counts the number of type-1 couplings with original data whose circuit diagram is shown in Figure 4. Level-1 AND gates of CNT1_1 detects the transitions as discussed in the above section. The outputs of level-1 AND gates are fed to OR gates. Type-1 coupling occurs when the first line is having transition and third line is idle and vice-versa, which clearly assures that these lines, must be inserted as inputs to an XOR gate to verify this condition. These five OR gate outputs are divided into 3 groups i.e., O0 and O2; O1 and O3; O2 and O4 which are fed to three different XOR gates. The outputs of these three XOR gates should be added using a full adder. This method implements the full adder using 8 transistors and output of the full adder represents the number (as there are four in number two bits are sufficient to represent the count) of Type-1 couplings K1K0.

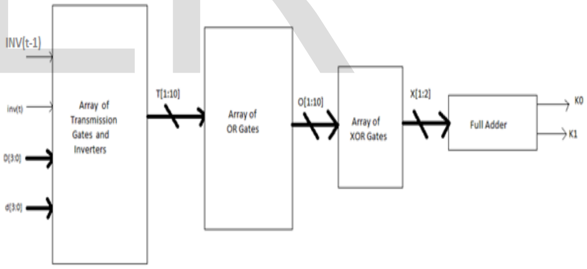

Figure 4: Circuit diagram of CNT1_1

CNT1_2: CNT1_2 counter counts the number of type-1 couplings with the inverted data. The inputs of CNT1_2 are $(\overline{d(t)},\overline{inv(t)})$ and (D(t-1), INV(t-1)) whereas the output is L1L0. The circuit diagram of this counter is shown in Figure 5.

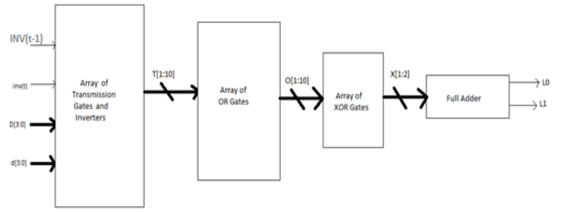

Figure 5: Circuit diagram of CNT1_2

XOR Stack: When either of N1 or N0 is 'high', the output of OR gate i.e., INV(t) becomes 'high' indicating a Type-1 or Type-0 switching condition. The truth table of XOR stack is shown in Table 3. The original data is transmitted only if both N1 and N0 are 'low'. For all other cases inverted data is transmitted to eliminate crosstalk.

Table 3: Truth table of XOR Stack

N1	N0	OR gate output ((INV(t))	Encoded Data (D(t),INV(t))
0	0	0	(D(t),0)
1	0	1	($\overline{D(t)}$,1)
0	1	1	($\overline{D(t)}$,1)
1	1	1	($\overline{D(t)}$,1)

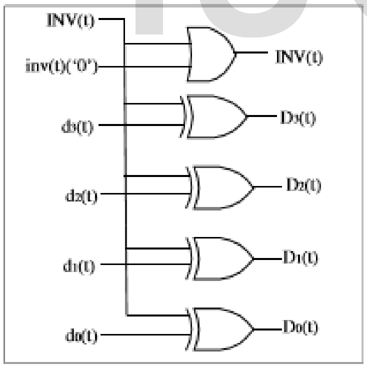

Figure 6: XOR Stack

Latch: As proposed design is comparing the present data with the previously transmitted data there is a necessity for storing the previously transmitted data. For this purpose latches are necessary that are implemented using transmission gates.

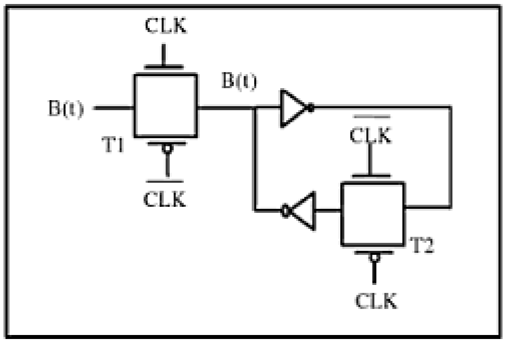

Figure 7: Latch using transmission gate

2-bit Comparator: The next block is a 2-bit comparator which compares the count of two type-1 counters (i.e., CNT1_1 and CNT1_2). The internal logic diagram is shown in Fig.6. After comparing K1K0 and L1L0, it generates an output N1 as logic 'high' when the count of CNT1_1 is greater than CNT1_2 whereas logic 'low' in the converse situation. The output of 2-bit comparator is fed as one of the input of the OR as shown in the block diagram Figure-8.

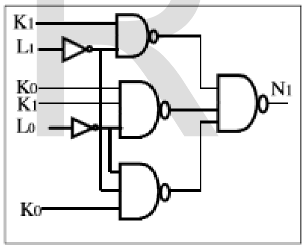

Figure 8: The 2-bit Comparator

For Type-4 and Type-3: This method consists four major blocks are Transition Detector, Type-4 Detector, Type-3 Detector and Multiplexer. The first block is the transition detector which compares present data bit with the previous data and then detects whether there is any transition present or not. After the detection of transition the next step is to see whether this transitions causes crosstalk or not for this purpose the method uses Type-4 detector to detect the Type-4 couplings and Type- 3 detector checks whether they cause the Type-3 coupling or not. If either of coupling present the pin became 'High'

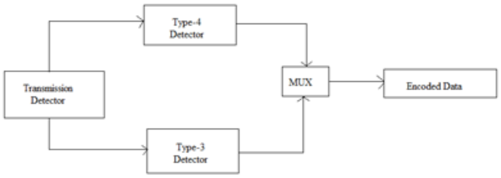

Figure 9: Block diagram of an encoder

Transition Detector: Transition detector checks whether there is any transition occur or not by using AND gates. The top five transmission gates detects the low to high transition and the bottom five transmission gates detects the high to low transition as shown in Figure 10. For this purpose it uses the data which is transmitted previously, it compares the present data with the previous data. If there is any transition the output becomes 'High' otherwise it is 'Low'.

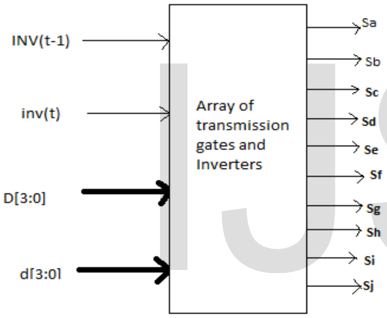

Figure 10: Block diagram of a Transition Detector

Type-4 Detector: As discussed in the above section there are two cases which cause Type-4 couplings are tabulated in Table 2. Now these cases which cause crosstalk are to be detected and that data which cause crosstalk is to be complemented to avoid crosstalk. There are five lines i.e. four data lines & one control line that makes the three combinations of lines i.e. (Sa, Sb, Sc), (Sb, Sc, Sd) and (Sc, Sd, Se) as shown in Figure 11. Depending on the cases the transition which cause crosstalk are given to the transmission gate to detect the case and after that all these are ORed so that if one of the case cause crosstalk then it will be at logic 'High'. Here OR gate is implemented by using NAND gate.

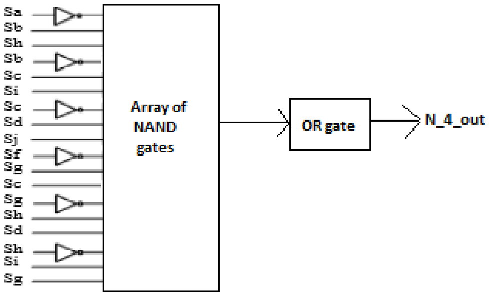

Figure 11: Block diagram of a Type-4 Detector

Type-3 Detector: As discussed in above section there are four cases which cause Type-3 couplings are tabulated in the Table 2. Now these cases which cause crosstalk are to be detected and that data which cause crosstalk is to be complemented to avoid crosstalk. There are five lines i.e. four data lines & one control line that makes the three combination of lines i.e. (Sa, Sb, Sc), (Sb, Sc, Sd), (Sc, Sd. Se) as shown in Figure 12.

Depending on the cases the transition which cause crosstalk are given to an AND gate to detect the case and after that all these are ORed so that if one of the case cause crosstalk then it will be at logic 'High' otherwise it became at logic 'Low'.

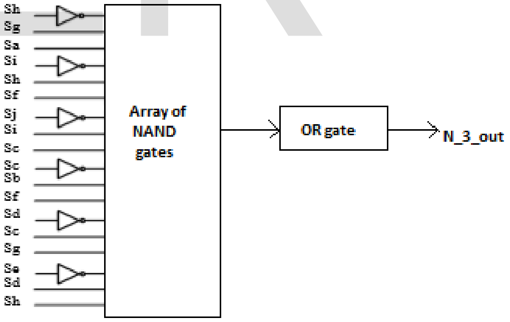

Figure 12: Block diagram of a Type-3 Detector

Multiplexer: The truth table of multiplexer is shown in Table 4. When either N_4_out or N_3_out is 'High', the inverted data must be transmitted otherwise original data bit is transmitted.

Table 4: Truth table of a Multiplexer

N4 out	N3 out	MUX OUTPUT
0	0	(D(t),0)
1	0	$(\overline{D(t)},1)$
0	1	$(\overline{D(t)},1)$
1	1	$(\overline{D(t)},1)$

In this the data bit is fed as one of the input for the XOR gate and control line is given as the second input for the 2-input XOR gate as shown in Figure 13. If the line is 'High' then it indicates that the inverted data must be transmitted and must be inverted to avoid the crosstalk and if it is 'Low' which indicates that the original data is to be transmitted.

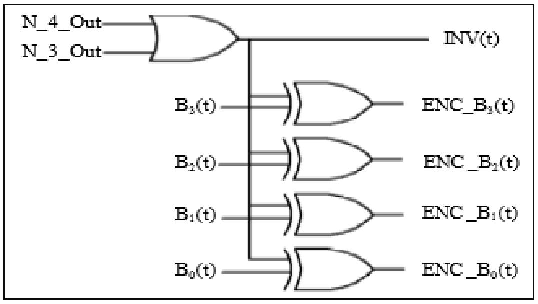

Figure 13: Multiplexer

Decoder: The function of decoder is to decode the encoded data. The internal circuit of decoder is shown in Figure 14.

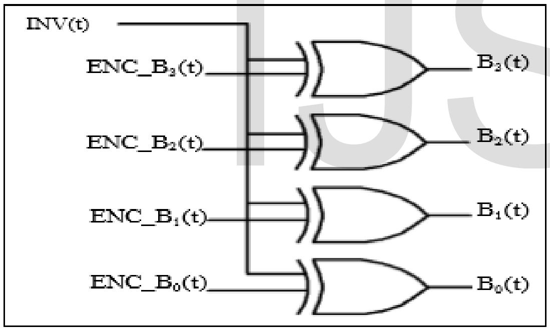

Figure 14: Decoder

The encoded data is fed as one of the input for the XOR gate and control line. (INV(t)) is given as second input for the 2-input XOR gate. If the (INV(t)) line is 'High' then it indicates that the inverted data has been transmitted and (ENC(t)) must be inverted to get the original data and if it is 'Low' which indicates that the original data has been transmitted.

In this proposed paper to reduce the power dissipation, crosstalk and the area of the chip size, all the XOR gates are implemented using three transistors and the full adder is implemented using eight transistors with the proper sizing of all the transistor.

The block diagram of xor using 3T and full adder using 8T is show in the figure 15 and figure 16 respectively.

8T Full Adder method is based on the use of a simple cell as shown in figure 16. XOR and XNOR functions are the key variables in adder equations. If the generation of them is optimized, this could greatly enhance the performance of the full adder cell.

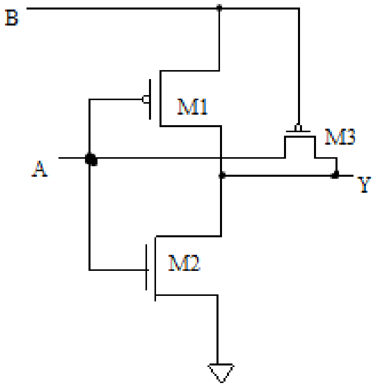

Figure 15: 3T XOR

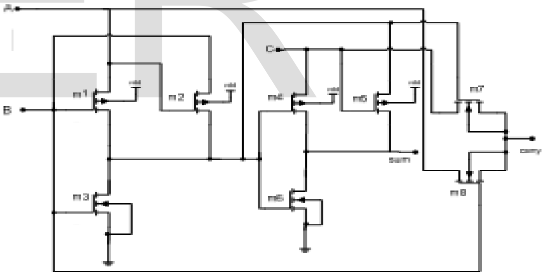

Figure 16: Full adder with 8T structure

A typical Full Adder in 8T logic as shown in figure 16 embodies only 8 transistors and the number of interconnections between them is highly reduced. Having each transistor a lower interconnection capacitance, the W/L can be close to the minimum value and the power consumption is decreased.

4. RESULT

The proposed encoder is simulated to find power dissipation of the codec. Simulation results were obtained using Cadence Virtuoso in 90nm technology.

The power dissipation and crosstalk delay of the circuitry (Encoder, Interconnects and Decoder) are obtained. Total power dissipation includes the power dissipated in encoder, interconnects and decoder (i.e. $P_{enc} + P_{dec} + P_{interconnect)}$. $P_{Decoded}$ is the dynamic power dissipation (which depends on the switching activities of the data) of the encoded data.

The crosstalk effect is reduced by inverting the original data and which in turn also reduces the switching activity. Here, the proposed method reduces both the Type-0, Type-1, Type-3 and Type-4 coupling. Type-0 and Type-1 coupling is caused in two and eight cases respectively. Therefore, switching activity reduces by 40.7% which is substantially more as compared to Fan et al [5].

Our proposed method has greatly reduced the chip area by the reduction in number of transistor as compared to [10] and [11] RC modelled. The proposed model occupies 30% lesser chip area than that of [10] and [11].

Table 5: Comparison of proposed model with [10] and [11]

Coding Method	Area(Number of transistors used)
Fan et.al	664
G.Nagendra Babu , Deepika Agarwal, B.K.Kaushik and S.K.Manhas(Type-0 and Type-1)	472
Deepika Agarwal, G. Nagendra Babu, B. K. Kaushik,S. K. Manhas(Type-3 and Type-4)	242
Proposed(Type-0 and Type-1)	334
Proposed(Type-3 and Type-4)	170

5. CONCLUSION

This paper demonstrated the reduction in crosstalk and power dissipation by using bus- invert method. The proposed model is having less number of transistors which decreases the chip area. The results show a reduction in circuit area by 30% compared to previously available models for capacitively modelled interconnects.

6. REFERENCES

[1] L. Trevillyan, D. Kung, R. Puri, L.N. Reddy, and M.A. Kazda, “An integration environment for technology closure of deep-submicron IC designs,” IEEE Des. Test. Comput. , vol. 21, no. 1, pp. 14- 22, 2004.

[2] L. He, and K.M. Lepak, “Simultaneous Shield Insertion and Net ordering for Capacitive and Inductive Coupling Minimization, ” Int. Symp. Physical Design , pp. 55-60, 2000.

[3] T. Shang-Wie, C. Yao-Wen, and J. Jing-Yang, “ RLC Coupling-Aware Simulation and On-Chip BusEncoding for Delay Reduction,” IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 10, 2006.

[4] K. Hirose, and H. Yasuura, “A bus delay reduction technique considering crosstalk,” In Proc. Design Automation and Test Eur. (DATE), Paris, France, pp. 441-445, 2000.

[5] C. Peng Fan, and C. Hao Fang, “Efficient RC low-power bus encoding methods for crosstalk reduction,” Integration, the VLSI Journal, Elsevier, vol. 44, no. 1, pp. 75-86, 2011.

[6] B. Victor, and K. Keutzer, “Bus encoding to prevent crosstalk delay,” in Proc. Int. Conf. on Computer-Aided Design, 2001, pp. 57-63.

[7] M. R. Stan, and W. P. Burleson, “Bus-Invert Coding for Low-Power I/O,” IEEE Trans. VLSI Syst., vol. 3, pp. 49-58, 2005.

[8] G. Nagendrababu, DeepikaAgarwal, B.K. Kaushik, S.K .Manhas, “Power and Crosstalk Reduction using Bus Encoding Technique for RLC Modeled VLSI Interconnect,” 2nd International Workshop on VLSI , 2011 , pp. 424–434, 2011.

[9] Chih-Peng Fan, and Chia-Hao Fang, “Efficient RC low-power bus encoding methods for crosstalk reduction,” Integration VLSI Journal, Elsevier, vol. 44, no. 1, pp. 75-86, Jan. 2011.

[10] G. NagendraBabu , DeepikaAgarwal , B. K. Kaushik and S. K. Manhas“Bus encoder for crosstalk avoidance in RLC modelled interconnects”, International journal of VLSI design & communication systems vol3,No.1, Febrary 2012.

[11] DeepikaAgarwal, G. NagendraBabu, B. K. Kaushik,S. K. Manhas “Reduction of Crosstalk in RC Modeled Interconnects with Low Power Encoder”978-1-4577-0240-2/11/$26.00 ©2011 IEEE.